

METHOD AND DEVICE FOR ALLOCATING CALL PROCESSING CPU RESOURCE

Publication number: JP7336736 (A)

Publication date: 1995-12-22

Inventor(s): ISODA KUNIIHIKO; NISHIBE YOSHIYASU; SAWADA TATSUO +

Applicant(s): NIPPON TELEGRAPH & TELEPHONE +

Classification:

- **international:** **G06F15/16; G06F15/177; H04Q3/545; G06F15/16; H04Q3/545;** (IPC1-7): G06F15/16; H04Q3/545

- **European:**

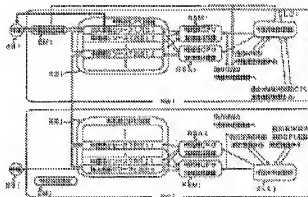
Application number: JP19940126633 19940608

Priority number(s): JP19940126633 19940608

Abstract of **JP 7336736 (A)**

PURPOSE: To obtain a method for minimizing a call loss rate in a multi-network and its device.

CONSTITUTION: A call traffic measurement device CMI measures a traffic from a call source SBI and a call processing CPU resource measurement device RSMi measures the allocation of a call processing CPU resource in a sub network NWi to a subscriber system and each relay system at a current point of time. Then a call loss rate measurement device CLCi calculates an average call loss rate of the entire multi-network based on the allocation of the call processing CPU resource and the traffic. The allocation of a call processing CPU resource in the sub network NWi to the subscriber system and each relay system minimizing the average call loss rate is calculated and a call processing



CPU resource allocation device RSAi allocates the call processing CPU resource of each sub network to the subscriber system and each relay system according to the calculated allocation.

Data supplied from the **espacenet** database — Worldwide

特開平7-336736

(43) 公開日 平成7年(1995)12月22日

(51) Int.Cl. ⁶	識別記号	庁内整理番号	F I	技術表示箇所
H 0 4 Q 3/545				
G 0 6 F 15/16	3 8 0 Z			
	4 7 0 S			

審査請求 未請求 請求項の数2 O L (全 8 頁)

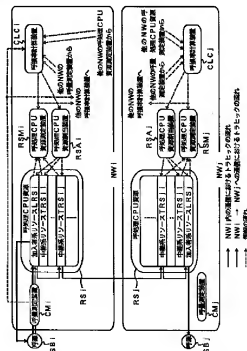
(21) 出願番号	特願平6-126633	(71) 出願人	000004226 日本電信電話株式会社 東京都新宿区西新宿三丁目19番2号
(22) 出願日	平成6年(1994)6月8日	(72) 発明者	磯田 邦彦 東京都千代田区内幸町1丁目1番6号 日 本電信電話株式会社内
		(72) 発明者	西部 喜康 東京都千代田区内幸町1丁目1番6号 日 本電信電話株式会社内
		(72) 発明者	澤田 立夫 東京都千代田区内幸町1丁目1番6号 日 本電信電話株式会社内
		(74) 代理人	弁理士 吉田 精孝

(54) 【発明の名称】 呼処理CPU資源割当方法及びその装置

(57) 【要約】

【目的】 マルチネットワークにおける呼損率を最小化し得る方法及びその装置を提供する。

【構成】 呼量測定装置C M iにより呼源S B iからの呼量を測定し、呼処理C P U資源測定装置R S M iによりサブネットワークN W i内の呼処理C P U資源の現時点における加入者系及び各中継系への割当を測定し、呼損率測定装置C L C iにより前記呼量及び呼処理C P U資源の割当に基づいてマルチネットワーク全体の平均呼損率を計算し、該平均呼損率が最小となる各サブネットワーク内の呼処理C P U資源の加入者系及び各中継系への割当を算出し、これに従って呼処理C P U資源割当装置R S A iによりサブネットワーク毎に呼処理C P U資源の加入者系及び各中継系への割当を行うようになした。



【特許請求の範囲】

【請求項1】 マルチネットワークを構成する複数の各サブネットワーク内の呼処理CPU資源の加入者系及び各中継系への割当を、各サブネットワークに収容される少なくとも1つの呼源で発生する呼、各サブネットワークの状態及びサブネットワーク数の変化に応じて決定する呼処理CPU資源割当方法において、各呼源からの呼の生起率及び平均保留時間を測定し、各サブネットワーク内の呼処理CPU資源の現時点における加入者系及び各中継系への割当を測定し、前記呼の生起率及び平均保留時間、呼処理CPU資源の割当に基づいてマルチネットワーク全体の平均呼損率を計算し、マルチネットワーク全体の平均呼損率が最小となる各サブネットワーク内の呼処理CPU資源の加入者系及び各中継系への割当を算出し、該算出した各サブネットワーク内の呼処理CPU資源の加入者系及び各中継系への割当に従って、サブネットワーク毎に呼処理CPU資源の加入者系及び各中継系への割当を行うようにしたことを特徴とする呼処理CPU資源割当方法。

【請求項2】 マルチネットワークを構成する複数の各サブネットワーク内の呼処理CPU資源の加入者系及び各中継系への割当を、各サブネットワークに収容される少なくとも1つの呼源で発生する呼、各サブネットワークの状態及びサブネットワーク数の変化に応じて決定する呼処理CPU資源割当装置において、各呼源毎に、呼の生起率及び平均保留時間を測定する呼量測定装置を設け、各サブネットワーク毎に、呼処理CPU資源の現時点における加入者系及び各中継系への割当を測定する呼処理CPU資源測定装置と、前記呼の生起率及び平均保留時間、呼処理CPU資源の割当に基づいてマルチネットワーク全体の平均呼損率を計算し、マルチネットワーク全体の平均呼損率が最小となる各サブネットワーク内の呼処理CPU資源の加入者系及び各中継系への割当を算出する呼損率計算装置と、該算出された各サブネットワーク内の呼処理CPU資源の加入者系及び各中継系への割当に従って、呼処理CPU資源の加入者系及び各中継系への割当を行う呼処理CPU資源割当装置とを設けたことを特徴とする呼処理CPU資源割当装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、少なくとも1つの呼源を収容したサブネットワークを複数個、相互に接続してなるマルチネットワークにおいて、呼損率を最小化し得る方法及びその装置に関するものである。

【0002】

【従来の技術】マルチネットワークにおける呼損率を向

上する方法として、従来、設備故障を含めたトラヒック変動に対し、(1)地域網統制御装置や加入者網統制御装置によりトラヒック制御を行う方法や、(2)設備を多重化する方法があった。

【0003】

【発明が解決しようとする課題】しかしながら、前者の方法では、(1)マルチネットワークに入力される呼を制限するため、呼損率が増加するという問題があった。また、後者の方法では、(2)マルチネットワークの状態及びトラヒック特性の長期的な予測を行って設備を建設する必要があるが、マルチネットワークの状態及びトラヒック特性の長期的な予測は困難であり、また、(3)予測と実際のトラヒック特性との間に差異が生じた場合に設備建設の誤りを生む結果となり、設備の余剰あるいは不足を招き、経済的でないという問題があった。

【0004】本発明の目的は、マルチネットワークに入力される呼の制限を必要とせず、マルチネットワークにおける呼損率を最小化し得る方法及びその装置を提供することにある。

【0005】また、本発明の他の目的は、マルチネットワークの状態及びトラヒック特性の長期的な予測を必要とせず、マルチネットワークにおける呼損率を最小化し得る方法及びその装置を提供することにある。

【0006】また、本発明のさらに他の目的は、設備の余剰や不足を招くことなく、マルチネットワークにおける呼損率を最小化し得る方法及びその装置を提供することにある。

【0007】

【課題を解決するための手段】本発明では、前記目的を達成するため、各サブネットワークが所有する呼処理のための設備、即ち少なくとも1個設置される呼処理CPUによって処理可能な全呼処理能力（以下、呼処理CPU資源と称す。）のうち、加入者系へ配分する割当及び各中継系へ配分する割当を、各サブネットワークに収容される少なくとも1つの呼源で発生する呼、各サブネットワークの状態及びサブネットワーク数の変化に応じて動的に変更することにより、マルチネットワークにおける呼損率を最小化するようにした。

【0008】なお、前述した呼処理CPU資源の割当は、複数の呼処理CPUのうちの何個を加入者系へ割り当て、何個を中継系のそれぞれへ割り当てるという、ハードウェア的な割当によって行う場合と、呼処理CPUを加入者系の呼処理CPUとして用いる時間の単位時間当たり（以下、中継系のそれぞれへ割り当てる割合と、時間（タイムシェアリング）的な割当によって行う場合と、これらを組み合わせた場合との3通りのやり方があり、本発明はこれらのいずれの場合も含むものとする。

【0009】具体的には、本発明では、マルチネットワ

ークを構成する複数の各サブネットワーク内の呼処理CPU資源の加入者系及び各中継系への割当を、各サブネットワークに収容される少なくとも1つの呼で発生する呼、各サブネットワークの状態及びサブネットワーク数の変化に応じて決定する呼処理CPU資源割当方法において、各呼源からの呼の生起率及び平均保留時間を測定し、各サブネットワーク内の呼処理CPU資源の現時点における加入者系及び各中継系への割当を測定し、前記呼の生起率及び平均保留時間、呼処理CPU資源の割当に基づいてマルチネットワーク全体の平均呼損率を計算し、マルチネットワーク全体の平均呼損率が最小となる各サブネットワーク内の呼処理CPU資源の加入者系及び各中継系への割当を算出し、該算出した各サブネットワーク内の呼処理CPU資源の加入者系及び各中継系への割当に従って、サブネットワーク毎に呼処理CPU資源の加入者系及び各中継系への割当を行うようになした。

【0010】また、本発明では、マルチネットワークを構成する複数の各サブネットワーク内の呼処理CPU資源の加入者系及び各中継系への割当を、各サブネットワークに収容される少なくとも1つの呼で発生する呼、各サブネットワークの状態及びサブネットワーク数の変化に応じて決定する呼処理CPU資源割当装置において、各呼源毎に、呼の生起率及び平均保留時間を測定する呼量測定装置を設け、各サブネットワーク毎に、呼処理CPU資源の現時点における加入者系及び各中継系への割当を測定する呼処理CPU資源測定装置と、前記呼の生起率及び平均保留時間、呼処理CPU資源の割当に基づいてマルチネットワーク全体の平均呼損率を計算し、マルチネットワーク全体の平均呼損率が最小となる各サブネットワーク内の呼処理CPU資源の加入者系及び各中継系への割当を算出する呼損率計算装置と、該算出された各サブネットワーク内の呼処理CPU資源の加入者系及び各中継系への割当に従って、呼処理CPU資源の加入者系及び各中継系への割当を行う呼処理CPU資源割当装置とを設けた。

【0011】

【作用】本発明方法によれば、各呼源で発生する呼の生起率及び平均保留時間が測定され、呼処理CPU資源の現時点における加入者系及び各中継系への割当が測定され、マルチネットワーク全体の平均呼損率が計算され、これを最小とする呼処理CPU資源の割当が算出され、各サブネットワーク内の呼処理CPU資源の加入者系及び各中継系への割当が行われる。

【0012】また、本発明装置によれば、各呼量測定装置により各呼源で発生する呼の生起率及び平均保留時間が測定され、呼処理CPU資源測定装置により呼処理CPU資源の現時点における加入者系及び各中継系への割当が測定され、呼損率計算装置によりマルチネットワーク全体の平均呼損率が計算され、これを最小とする呼処

理CPU資源の割当が算出され、呼処理CPU資源割当装置により各サブネットワーク内の呼処理CPU資源の加入者系及び各中継系への割当が行われる。

【0013】

【実施例】以下、本発明の実施例を図面に基づいて説明する。

【0014】図1は本発明の呼処理CPU資源割当方法の概要を示すもので、ここではメッシュ状態に接続された3つのサブネットワークからなるマルチネットワークの例を示す。図中、NW1、NW2、NW3はサブネットワーク、SB1、SB2、SB3は呼源、LRS1、LRS2、LRS3は加入者系呼処理CPU資源(リソース)、TRS12、TRS13、TRS21、TRS22、TRS23、TRS31、TRS32は中継系呼処理CPU資源(リソース)である。

【0015】前記加入者系リソースLRS1～LRS3は、それぞれ各サブネットワークNW1～NW3内の呼処理CPU資源のうち加入者系へ配分された割当のことであり、それぞれ各サブネットワークNW1～NW3に収容された呼源SB1～SB3に対する呼処理を行う。

【0016】また、前記中継系リソースTRS12及びTRS13はサブネットワークNW1内の呼処理CPU資源のうち中継系へ配分された割当のことであり、それぞれサブネットワークNW2及びNW3に対する呼処理を行う。また、中継系リソースTRS12及びTRS32はサブネットワークNW2内の呼処理CPU資源のうち中継系へ配分された割当のことであり、それぞれサブネットワークNW1及びNW3に対する呼処理を行う。また、中継系リソースTRS31及びTRS32はサブネットワークNW3内の呼処理CPU資源のうち中継系へ配分された割当のことであり、それぞれサブネットワークNW1及びNW2に対する呼処理を行う。

【0017】図2はサブネットワーク内の詳細を示すもので、ここではN個のサブネットワークのうちの1番目及びj番目のサブネットワークについて示す。即ち、図中、NW_i、NW_jはサブネットワーク、SB₁、SB_jは呼源、RS_i、RS_jは呼処理CPU資源、LRS_i、LRS_jは加入者系リソース、TRS₁₁、……、TRS_{1j}、……及びTRS_{j1}、……、TRS_{jj}、……は中継系リソース、CM₁、CM_jは呼量測定装置、RSM₁、RSM_jは呼処理CPU資源測定装置、CLC₁、CLC_jは呼損率計算装置、RSA_i、RSA_jは呼処理CPU資源割当装置である。

【0018】呼処理CPU資源RS_i、RS_jは、それぞれサブネットワークNW_i、NW_jが所有する呼処理CPUによって処理可能な全呼処理能力を表している。また、加入者系リソースLRS_i、LRS_jは、それぞれ前記呼処理CPU資源RS_i、RS_jのうち加入者系へ配分された割当を表している。また、中継系リソースTRS₁₁、……、TRS_{1j}、……及びTRS_{j1}、……、TRS_{jj}

Sji, ……は、それぞれ前記呼処理CPU資源Rsi, RSjのうち各中継系へ配分された割当を表している（なお、符号中の2桁の数字のうち、1桁目は該当サブネットワークの番号を、2桁目は呼処理を行う中継先のサブネットワークの番号を表している。）。

【0019】呼量測定装置Cmi, CMjは、それぞれ呼源Sbi, SBjから発生する呼の生起率及び平均保留時間を測定し、該測定結果を全てのサブネットワークの呼損率計算装置へ送出する。なお、ここでは各サブネットワーク毎に呼源が1個のみ取容されている場合について示したが、通常は複数の呼源を取容しており、その場合は各呼源毎に呼量測定装置が設けられる。

【0020】呼処理CPU資源測定装置RSMi, RSMjは、それぞれ呼処理CPU資源Rsi, RSjの現時点における加入者系及び各中継系への割当を測定し、該測定結果を全てのサブネットワークの呼損率計算装置へ送出する。

【0021】呼損率計算装置CLCi, CLCjは、前記送られてきた呼の生起率及び平均保留時間、呼処理CPU資源の割当に基づいてマルチネットワーク全体の平均呼損率を計算し、マルチネットワーク全体の平均呼損率が最小となるサブネットワークNW i, NW j内の呼処理CPU資源Rsi, RSjの加入者系及び各中継系への割当を算出し、これを呼処理CPU資源割当装置RSA i, RSA jへ送出する。

【0022】呼処理CPU資源割当装置RSA i, RSA jは、それぞれ前記送られてきたサブネットワークNW i, NW j内の呼処理CPU資源Rsi, RSjの加入者系及び各中継系への割当に従って、呼処理CPU資源Rsi, RSjの加入者系及び各中継系への割当を行う。

$$B_{ij} = \frac{(\alpha_{ij})^{n_{ij}}}{n_{ij}!} \cdot \frac{n_{ij}}{\sum_{k=0}^{\infty} \frac{(\alpha_{ij})^k}{k!}} \quad \text{……(1)}$$

$$\alpha_{ij} = \sum_{k=1}^n \lambda_{ijk} b_{ik} + \sum_{k=1}^n (1 - \delta_{ik}) (1 - B_{ik}) (1 - B_{jk}) \lambda_{kij} h_{ki} \quad \text{……(2)}$$

$$\alpha_{ij} = (1 - B_{ij}) \lambda_{ij} h_{ij} + (1 - B_{ji}) (1 - B_{jj}) \lambda_{jij} h_{ji} \quad \text{……(3)}$$

ここで、 α_{ij} は、 $i = j$ の場合は、サブネットワークNW i内の加入者系リソースLRS iに入ってくる呼量の総和を表し、 $i \neq j$ の場合は、サブネットワークNW i内のサブネットワークNW jに対する中継系リソースTRSi jに入ってくる呼量の総和を表す。

【0028】また、 B_{ij} は、 $i = j$ の場合は、サブネットワークNW i内の加入者系リソースLRS iでの呼損

【0023】以下、前述した測定及び計算について説明するが、パラメータは、サブネットワーク数N[個]、生起率 λ_{ij} [call/s]、平均保留時間 h_{ij} [s]、呼処理CPU資源 n_{ij} [a.u.: 任意単位]とする。ここで、 λ_{ij} は、サブネットワークNW iからサブネットワークNW jへの通信における呼の生起率を表し、 h_{ij} はサブネットワークNW iからサブネットワークNW jへの通信における呼の平均保留時間を表し、 n_{ij} は、 $i = j$ の場合は、サブネットワークNW i内の加入者系の呼処理CPU資源を表し、 $i \neq j$ の場合は、サブネットワークNW i内のサブネットワークNW jに対する中継系の呼処理CPU資源を表すものとする。

【0024】各呼量測定装置で測定された $\{\lambda_{ij}\}$ 、 h_{ij} 及び各呼処理CPU資源測定装置で測定された $\{n_{ij}\}$ の値は全ての呼損率計算装置に入力される。

【0025】各呼損率計算装置では入力された値を収集し、これらのデータを用いて、以下の処理を行うことにより、マルチネットワーク全体の平均呼損率 $B^{(K)}_{all}$ を計算する。

【0026】まず、 $\{n_{ij}\}$ の値を基に、全ての i ($i = 1, 2, \dots, N$) に対して、

【数1】

$$\sum_{j=1}^N n_{ij}$$

の値を計算し、 n_{ij} の割当配分を抽出する。各々の割当配分について、以下の(1)式から(3)式に示した連立方程式の解法をそれぞれ行う。

【0027】

【数2】

率を表し、 $i \neq j$ の場合は、サブネットワークNW i内のサブネットワークNW jに対する中継系リソースTRSi jでの呼損率を表す。

【0029】また、 δ_{ij} はクロネッカーのデルタを表し、

【数3】

$$\delta_{ij} = \begin{cases} 1 & (i=j) \\ 0 & (i \neq j) \end{cases} \quad \cdots(4)$$

である。

【0030】(1) 式から(3) 式で示した連立方程式の解 $\{\alpha_{ij}, B_{ij}\}$ ($i, j=1, 2, \dots, N$) のうち、 $\{B_{ij}\}$ を用いて、(5) 式で定義するマルチネットワ

ク全体の平均呼損率 $B^{(N)}_{all}$ を計算する。

【0031】

【数4】

$$B^{(N)}_{all} = \sum_{i=1}^N \sum_{j=1}^N \left[\{ (i \rightarrow j) \text{ 通信の比率} \} \right. \\ \left. \times \{ (i \rightarrow j) \text{ 通信における呼損率} \} \right] \\ = \sum_{i=1}^N \sum_{j=1}^N [\delta_{ij} \lambda_{ij} B_{ij} + (1 - \delta_{ij}) \lambda_{ij} \{ (1 - (1 - B_{ij}) \\ (1 - B_{ij}) (1 - B_{ij}) (1 - B_{ij}) \}] / \sum_{i=1}^N \sum_{j=1}^N \lambda_{ij} \quad \cdots(5)$$

呼処理CPU資源の割当配分の仕方に応じて、(5) 式で示した平均呼損率は様々な値をとるが、その中で、平均呼損率の値を最小値とする呼処理CPU資源の割当配分を実際に呼処理CPU資源として割り当てる値として決定する。

【0032】このマルチネットワーク全体の平均呼損率の計算から、平均呼損率の値を最小値とする呼処理CPU資源の割当配分の決定までの処理は、呼損率計算装置で行われる。図3に呼損率計算装置の詳細を示す。ここでは、大別して、データ収集部と計算処理部とに分けられる。

【0033】データ処理部では、呼量測定装置から送られてくるトラヒック特性 $\{\lambda_{ij}, h_{ij}\}$ 及び呼処理CPU資源測定装置から送られてくる現時点で割り当てられている呼処理CPU資源 $\{n_{ij}\}$ のデータを収集し、これらのデータを計算処理部に送る。計算処理部では、データ処理部から送られてきたデータ $\{\lambda_{ij}, h_{ij}, n_{ij}\}$ を用いて、マルチネットワーク全体の平均呼損率を計算する。

【0034】前述した呼損率計算装置における処理のフローを図4に示す。

【0035】次に、本発明を評価するために行った解析結果について説明する。ここでは、3つのサブネットワークから構成されるマルチネットワークを1つの例として解析した。呼源からの呼に関して、生起率は自サブネットワーク内通信の場合0.02 [call/s]、他サブネットワークへの通信の場合、各々に対し0.01 [call/s]、平均保留時間は自サブネットワーク内通信及び他サブネットワークへの通信共に100 [s]とした。

【0036】各サブネットワークが所有する呼処理CPU資源の総量を、10、20、30 [a.u.]の3通りの場合について、呼処理CPU資源の割当配分とその時のマルチネットワーク全体の平均呼損率との関係を図5に

示す。ここで、呼処理CPU資源の加入者系/中継系への割当比率 n_{11}/n_{13} は、各サブネットワークとも同一とした。また、各中継系に割り当てる呼処理CPU資源も等しくした。即ち、

$$n_{11} = n_{22} = n_{33} (= n_{11})$$

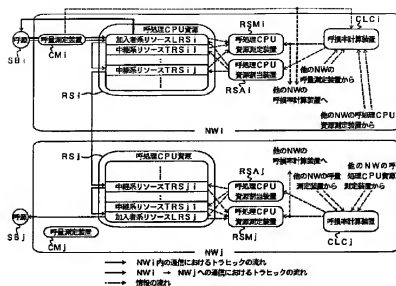
かつ $n_{12} = n_{13} = n_{21} = n_{23} = n_{31} = n_{32} (= n_{13})$ とした。

【0037】図5より、マルチネットワーク全体の平均呼損率の最小値を与える呼処理CPU資源の割当解が存在することが証明できる。また、最適な呼処理CPU資源の割当から外れていくにつれて、平均呼損率の値も大きくなることが分かる。解析結果から求められたマルチネットワーク全体の平均呼損率の最小値を与える呼処理CPU資源の割当解を図6に示す。

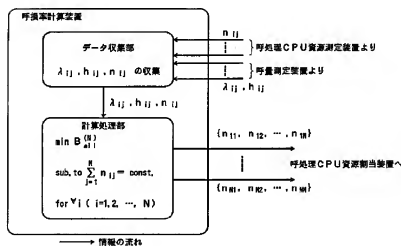
【0038】次に、マルチネットワークにおける、(1) 定常状態→(ii) 障害発生→(iii) 呼処理CPU資源の再割当後という状態遷移の中で、障害発生後に呼処理CPU資源を割り当てることによってマルチネットワーク全体の平均呼損率を改善できることを説明する。ここでも、3つのサブネットワークから構成されるマルチネットワークを1つの例として解析した。呼源からの呼に関して、生起率及び平均保留時間は前述した例と同じ条件とした。また、呼処理CPU資源の加入者系/中継系への割当配分は各サブネットワークで別々とした。

【0039】定常状態では、各サブネットワークが有する呼処理CPU資源の総量は10 [a.u.]とした。サブネットワークNW3内での障害発生により、該サブネットワークNW3においてサブネットワークNW1との中継系リソースとして割り当てられていた呼処理CPU資源の大きさが1 [a.u.] 減ったとする。その後、呼処理CPU資源の再割当を行うことを考えた。(i) 定常状態→(ii) 障害発生→(iii) 呼処理CPU資源の再割当後という状態遷移の中でのマルチネットワーク全体の平均呼

【図2】



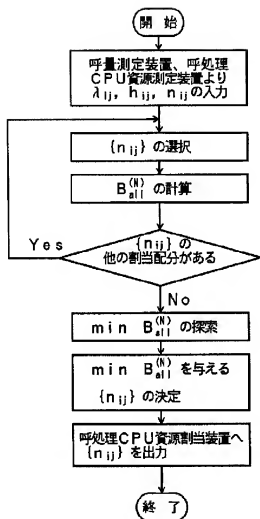
【図3】



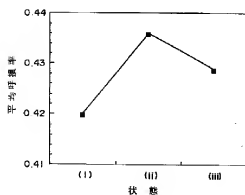
【図6】

$\sum_{j=1}^3 n_{ij}$ [a. u.]	呼出率CPU資源割当率		
	n_{11} [a. u.]	n_{12} [a. u.]	n_{13} [a. u.]
10	6	2	2
20	10	5	5
30	18	7	7

【図4】



【図7】



【図8】

	(i) 定常状態	(ii) 障害発生	(iii) 再創出後
n ₁₁	8	8	7
n ₁₂	2	2	2
n ₁₃	2	2	1
n ₂₁	2	2	2
n ₂₂	8	8	7
n ₂₃	2	2	1
n ₃₁	2	1	1
n ₃₂	2	2	1
n ₃₃	8	8	7
B ⁽²⁾ _{all}	0.420	0.436	0.429